

University of Groningen

INTER-LINE DISTANCE ESTIMATION AND TEXT LINE EXTRACTION FOR UNCONSTRAINED ONLINE HANDWRITING

Ratzlaff, E.

Published in:
EPRINTS-BOOK-TITLE

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2004

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):
Ratzlaff, E. (2004). INTER-LINE DISTANCE ESTIMATION AND TEXT LINE EXTRACTION FOR UNCONSTRAINED ONLINE HANDWRITING. In *EPRINTS-BOOK-TITLE* s.n..

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

INTER-LINE DISTANCE ESTIMATION AND TEXT LINE EXTRACTION FOR UNCONSTRAINED ONLINE HANDWRITING

EUGENE H. RATZLAFF

IBM T. J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598, USA
E-mail: ratzlaff@us.ibm.com

Methods for detecting and extracting whole text lines from unconstrained online handwritten text are described. The general approach is a “bottom-up” clustering of discrete strokes into small groups that are then merged into isolated lines of text. Initial clustering of strokes into groups is based on combined temporal and spatial stroke proximity. Spatial stroke proximity is gauged relative to estimated inter-line distance and mean character height. Two methods applicable to off-line or on-line data are described for estimating the inter-line distance: autocorrelation of the Y-axis projection histogram, and a fitting function. Inter-line distance is accurately determined for 99% of all text pages. Text line extraction accuracy on letters (correspondence) is 98.7% and on tables is 94.9%.

1 Introduction

One of the first steps often required in applying machine recognition to unconstrained handwriting is the identification and extraction of each line of text from among other lines. The purpose of text line extraction is to prepare data to meet the requirements of succeeding processing steps such as size normalization, word segmentation, or feature extraction. These steps typically require the data to be no more than a single row of characters. The goal of text line extraction is to assign correctly each stroke or component to its appropriate text line so that each isolated line may be passed in turn to the following analysis stage. The task is made difficult by the fact that data frequently contain undulations and shifts in the baseline, baseline skew, baseline-skew variability, character-size variability, sparse data, skipped lines and inter-line distance variability. Several of these issues have been addressed in much of the recent work focused on text line extraction for offline data [2,4,8].

Our interest has been in the area of recognition of unconstrained, online handwriting, particularly in support of personal digital notepad (PDN) devices, the IBM Ink Manager™ application, and the IBM Ink Software Development Kit (SDK) [11]. A PDN is a portable digitizer-and-pen device that electronically records the pen strokes while the user writes on a standard paper notepad. Each ink stroke is recorded as a sequence of (X, Y) points indexed in time. The primary role for the IBM unconstrained, online handwriting recognizer [5,9,10] in this context is delayed recognition of this dynamically recorded data for keyword and note-taking

transcription. In the past, we have constrained users to submit for transcription only portions of text written in strictly left-to-right, top-to-bottom order. We then relied on the fact that the desired spatial ordering follows from the temporal ordering so that text line extraction required only that we find apparent new-line events. We wished to implement a text line extraction scheme that removes this ordering constraint.

Since offline data contain neither the point-by-point vector properties within each distinct stroke nor the stroke-to-stroke temporal sequence, most of the earlier approaches to text line extraction have not addressed the use of such information. Our online data provides this temporal information, which can be distinctly valuable in several situations. For example, suppose a very small stroke is found in the space between two text lines as with the dot above *being* in Figure 1. The spatial evidence alone might suggest that the mark could just as easily be grouped with the upper text line as the lower, assuming it to be a low period or high dot, respectively. However, if the stroke is temporally bounded by two nearby strokes (here, *being* and fragment *res*), then the spatiotemporal evidence strongly suggests that all three strokes belong to the same text line, and the ambiguity is resolved. Conversely, if the mark is temporally bounded by the strokes from *finally* and *completed*, the stroke would be considered part of the upper line.

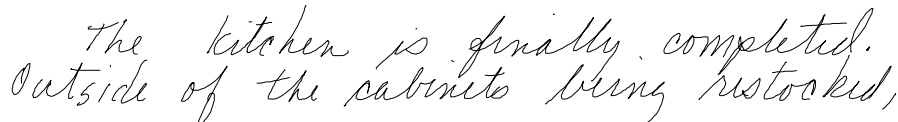


Figure 1. Temporal context resolves spatially ambiguous text line membership (see text).

Since it is intuitively clear that temporal cues should be useful for text line extraction, we are exploring methods that take advantage of both the spatial and temporal information. The general approach is a “bottom-up” clustering of discrete strokes into increasingly larger groups that eventually merge to complete text lines. The initial clustering is based on the strong evidence of spatiotemporal proximity. Subsequent merging is based on more sophisticated metrics that include dependencies on estimates of inter-line distance and mean character height.

Two methods for estimating the inter-line distance are described in the following section. One method first projects all strokes onto the Y-axis to create a histogram distribution. The autocorrelation of this projection profile is then evaluated to find the periodicity that reflects the likely inter-line distance. The other method finds the best inter-line distance fit by finding a distribution of evenly-spaced horizontal lines that best minimizes, over all strokes, the summed distances of all points in each stroke to the line nearest the stroke. Mean character height is estimated from conventional Y-axis projection profiles [1,3].

2 Methods

The bottom-up clustering method for text line extraction described herein proceeds in five principal stages. Several assumptions have been made owing to the nature of the application environment. As a rule, we anticipate that our users will use lined paper on their PDN, so we have initially assumed that the data will not have a large baseline skew nor have gross baseline undulation. We also assume that each ink stroke is indexed according to its original temporal sequence. All work has been done with handwriting in the English language, although this method and aspects thereof should be applicable to other languages.

2.1 *Assign descriptors to each stroke*

The initial processing step first assigns descriptors to each stroke. Descriptors include both exact measurements or metrics, such as the stroke centroid (center of gravity) and the stroke X and Y extrema, as well as heuristic estimates, such as to what extent the stroke appears to be a dot, straight line, or potential diacritical mark.

2.2 *Generate Forward Progression (FP) groups*

The Y-axis projection histogram is generated for each stroke, then the initial bottom-up clustering begins by creating Forward Projection (FP) groups. Strokes are merged into FP groups if they are temporally adjacent and have strongly overlapping Y-axis projections. A single, unmerged stroke becomes an independent FP group. The intent is to be very conservative by only merging strokes based on very strong spatiotemporal evidence; the process is not iterative. The value of FP grouping is twofold: to initiate bottom-up clustering and to create stroke assemblies that span multiple characters so as to assess mean character height. If FP groups become wide, they may also begin to misrepresent character height if baseline skew or undulation is significant. For this reason, wide FP groups are split into smaller groups when and if significant horizontal gaps exist between strokes.

2.3 *Estimate mean character height*

Y-axis projection histograms are created for each FP group. Each FP histogram is smoothed and the dominant peak's full width at half height (FWHH) is estimated for each FP histogram. The weighted average (weighted by the summed arc-lengths of the member strokes) of the FP histogram FWHH estimates is used as a metric of the overall mean character height.

2.4 Estimate inter-line distance

As described below, inter-line distance is estimated with projection profile autocorrelation. If the estimate of the reliability of the autocorrelation is not high, a fitting function is used to corroborate the autocorrelation estimate. If a single measure cannot be determined with confidence, the two estimates are combined to give an inter-line distance range.

2.4.1 Autocorrelation of the Y-axis projection histogram

All strokes, except those labeled with descriptors as likely dots or diacritics, are projected onto the Y-axis. The autocorrelation [7] of this projection is evaluated, typically over a range limited by an upper bound. This bound is the sum of the estimated mean character height, the “height” of the largest gap in the projection, and the height of the tallest stroke. When only a small amount of data is available (typically, fewer than 15-20 characters), it is also useful to externally introduce a rough estimate of the expected inter-line space to avoid limiting the upper bound of the autocorrelation. As shown in Figure 2, the autocorrelation gives rise to peaks that reveal the periodicity of the data. Ideally, the distance of the second peak from the origin (8.67mm in Figure 2) gives the inter-line spacing. The relative magnitude and cyclic regularity of the secondary and tertiary peaks and valleys, when present, are used to heuristically estimate the reliability of the measurement.

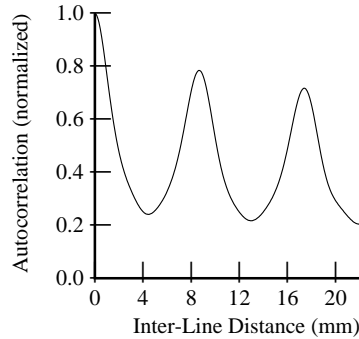


Figure 2. Autocorrelation of Letter data in Figure 4; ordinate data normalized to 1.0 at maximum.

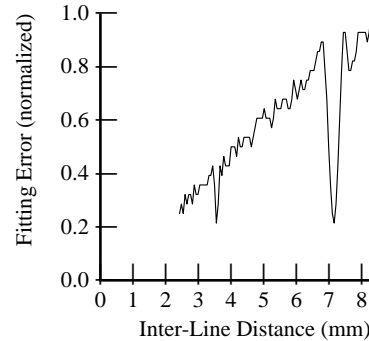


Figure 3. Fitting error of Tables data in Figure 4; ordinate data normalized to 1.0 at maximum.

2.4.2 Fitting function

The fitting process is somewhat analogous to sliding lined pages with different spacings up and down under the ink to find the page that best matches the ink spacing. A fitting error is defined as the weighted distance (weighted by the summed

arc-lengths of the FP group member strokes) from the centroid of a FP group to a hypothetical horizontal line. The fitting error between each FP group and the nearest line in a set of evenly spaced horizontal lines is summed over all FP groups. This is repeated for several offset displacements of the lines. The minimum sum fitting error over all offsets is considered the optimal fit for a given inter-line distance. Having found the lowest fitting error for a given inter-line distance, the process is repeated for a range of inter-line distances. As with the autocorrelation, the fit is only estimated over a limited range. The inter-line distance is typically found at the strongest minimum from baseline in the fitting error curve (7.15mm in Figure 3).

2.5 *Merge FP groups into text line groups*

FP groups are iteratively merged into complete text line groups in a best-first loop based on estimates of merge confidence. Computationally simple methods are applied first in assessing merge confidence, followed by more complex evaluations. At present, these estimates are heuristic and are programmatically tuned. The evidences found useful in these heuristics, for or against merging, are as follows:

- the strength of the overlap between the Y-axis FP group projections
- the distance between the Y-axis centroids of the two FP groups in relationship to the estimated mean character height and/or inter-line distance
- spatial, temporal and spatiotemporal relationships between strokes
- group magnitude (number, size, and extrema of all strokes in group)
- descriptor estimates for single-stroke groups (e.g. does it appear to be a dot?)

2.6 *Detect and subdivide text line groups into phrases*

Finally, text lines are subdivided into shorter phrases. The goal is to create phrases that do not split whole words, are neither too short nor too long for good baseline and normalization estimation, and are minimized to reduce multi-word recognition complexity. Phrases are created only where inter-stroke gaps are clearly large relative to the estimated mean character height.

3 **Experimental Results**

Data were collected from 118 writers at a resolution of 10 dots-per-mm using the *CrossPad*TM PDN (A.T. Cross Company) and IBM *Ink Manager*TM software. Writers were arbitrarily assigned either college rule or standard rule paper. Three different types of tabular data, each requiring an entry sequence that was either semi-random or column-by-column, and an original letter of correspondence were

elicited from each writer; these sets are henceforth referred to as the Tables and Letters sets, respectively (Figures 4,6,7). The median number of lines in the Letters was 14, the minimum 6; and for the Tables, 17 and 5, respectively.

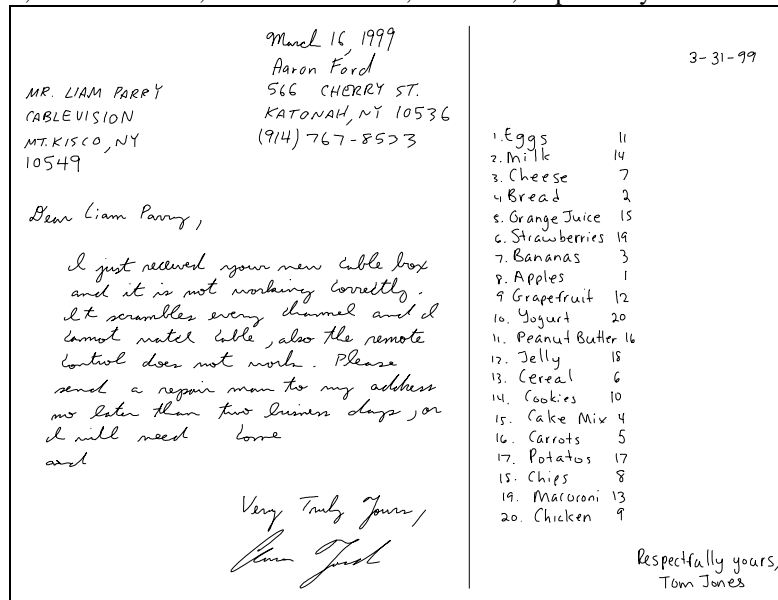


Figure 4. Example Letters and Tables pages with 18 and 23 text lines, respectively.

Inter-line distances were accurately estimated for all 114 Letters and for all but 4 of 327 Tables. Each of these four cases was a predominantly double-spaced page having 13-16 lines written, each text line preceded and followed by a blank line, with only one or two single-spaced exceptions (example, Figure 7a). In 3 of 4 instances the inter-line distances were reported as ranges (due to low confidence in the estimate) that suitably spanned the scale from single-spacing to double-spacing. In the fourth case, the inter-line distance was inaccurately evaluated to be twice the correct single-space distance. Figure 5 represents the combined inter-line distance estimates for 126 college rule pages (~7.1mm/line) and 310 standard rule pages (~8.7mm/line); the relative standard deviation for each peak is 1% in each case.

The true number of lines written on each page was determined by visual examination of the data in the absence of the original lined rulings. Table 1 shows the performance of the method for text line extraction. Text line splits are favored over text line merges by design because we feel this is less likely to cause confusions to the user. In calculating the word errors a word token was defined as a space-delimited symbol string or a new-line symbol. By visual inspection, the word error rate was estimated as the minimum number of operations required to correct

the result using an error count of 1 for each of the following operations: word deletion, word replacement, word insertion, text line deletion.

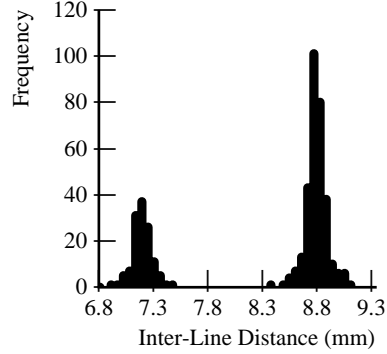


Figure 5. Histogram distribution of 436 correctly estimated inter-line distance estimates (one correct estimate of 17.4mm for a completely double-spaced Table page is excluded).

Table 1. Breakdown of data and errors attributable to text line clustering on Letter and Table data sets at the page, text line, and word level. Error rates are given as percentage of total.

	Pages		Text Lines			Words	
	Total	Flawed	Total	Lines Merged	Lines Split	Total	Error
Letters	114	16 14%	1608	6 0.4%	15 0.9%	10090	42 0.4%
Tables	327	118 36%	5291	39 0.7%	231 4.4%	16121	373 2.3%

4 Discussion and Future Developments

The overall performance of the system meets most expectations (Table 1; Figure 6), however, several problems were observed. Significant skew was found in several instances. As seen in Figure 7b, a mere 2° line skew can shift words at either end of a text line by one inter-line space, resulting in significant text line splitting. Despite the use of lined paper, undulate text and somewhat irregular line spacing is observed (Figure 7c); some of these irregularities apparently occurred when the paper occasionally shifted on the PDN. This was especially true when writers entered lists into tables or appended dates to the top of a block of text. Reliable inter-line distance estimates were made despite skew and shift problems; however, text line extraction performance occasionally degraded.

The application of spatiotemporal information is clearly advantageous for the accurate partitioning of dots, short strokes, delayed strokes and diacritical marks that are temporally bounded by other strokes in their own text line (Figure 6). Many of the errors that were observed occurred when the misassigned stroke was temporally at the beginning or end of a text line and the spatiotemporal information was ambiguous. This frequently occurred with short strokes landing high above or well below the baseline, especially at the start or end of a text line, as with the comma in phrases such as *Dear Tom*, or *Best regards*,. In Figure 7d, for example, four commas were inserted after four city names were written, resulting in a line split. Similar errors were more common in the Tables, where spatiotemporally adjacent strokes were sparse, as with the numbers in the right-most column of Figure 4, or where there were significant gaps between table entries in different columns.

Although several of the errors occurred with visually ambiguous data, there appears to be room for significant improvement. Skew detection and correction should be considered. Reliable inter-line distance estimates can be used to place upper bounds on the number of lines possible, enabling the detection and correction of gross splitting errors (Figures 7b, 7c). In addition, by replacing the many programmatically-tuned heuristics involved in the clustering process with a more probabilistic approach based on training data, the bottom-up clustering assignments should improve. The application of generalized projections [6] and local extrema points [4,8] hold promise for improved character height estimation and baseline estimation; these improved or added features could likely be applied to advantage.

January 21, 1999

Killian Yarns Company
424 Fifth Avenue
Mt. Vernon, NY 10807

Dear Mr. Vernon,

Two weeks ago, I received my order of a half dozen picture frames from your company. I'm sorry to say that three of the said frames were broken upon receipt. I am returning the frames in the hope of your at no cost to me since they were broken in the box. Also, I would like reimbursement for my shipping the frames back to you. Enclosed is the shipping receipt.

Very truly yours,
John Yarns
60 Forest Avenue
New Rochelle, NY 10801

Figure 6. Extraction results for page with appreciably overlapping text lines. Extracted text lines are shown alternately with normal and bold points from top; each of 20 lines was correctly extracted.

5 Acknowledgements

The author thanks Millie Miladinov, Michael Perrone, John Pitrelli, and Jay Subrahmonia of the IBM Pen Technologies Group for their contributions.

References

1. Bozinovic R. and Srihari S., Off-line cursive script word recognition, *IEEE Trans on PAMI*, **11** (10), January 1989, pp. 68-83.
2. Bruzzone E. and Coffetti M. C., An algorithm for extracting cursive text lines, *Proceedings of the Fifth International Conference on Document Analysis and Recognition ICDAR '99* Bangalore, India 20-22 Sept. 1999, pp. 749-752.
3. Ernest L. D., Machine recognition of cursive script, *Proc. IFIP Congress*, **62**, 1992, 462-466.
4. Kim G., Govindaraju V. and Srihari S. N., Architecture for handwritten text recognition system, *Proceedings 6th IWFHR*, Taejon, Korea, August 1998, pp. 113-122.
5. Nathan K. S., Beigi H. S. M., Subrahmonia J., Clary G. J. and Maruyama H., Real-time on-line unconstrained handwriting recognition using statistical methods, *Proceedings of ICASSP 95: IEEE International Conference on Acoustics, Speech, and Signal Processing*, Detroit, Michigan, May 8-12, 1995, **4**, pp. 2619-2622.
6. Nicchiotti G. and Scagliola C., Generalised projections: a tool for cursive handwriting normalization, *Proceedings of the Fifth International Conference on Document Analysis and Recognition ICDAR '99* Bangalore, India 20-22 Sept. 1999, pp. 729-732.
7. Press W. H., Teukolsky S. A., Vetterling W. T., and Flannery B. P., *Numerical recipes in C*, 2nd Ed. (Cambridge University Press, New York, 1992).
8. Pu Y. and Shi Z., A natural learning algorithm based on Hough transform for text lines extraction in handwritten documents, *Proceedings 6th IWFHR*, Taejon, Korea, August 1998, pp. 637-646.
9. Ratzlaff E. H., Nathan K. S. and Maruyama H., Search Issues in the IBM Large Vocabulary Unconstrained Handwriting Recognizer, *Proceedings 5th IWFHR*, Colchester, England, September 1996, pp. 177-182.
10. Subrahmonia J., Nathan K. S. and Perrone M., Writer dependent recognition of on-line unconstrained handwriting, *Proceedings of ICASSP 96: IEEE International Conference on Acoustics, Speech, and Signal Processing*, Atlanta, Georgia, May 7-11, 1996, **6**, pp. 3478-3481.
11. <http://www.research.ibm.com/electricInk>